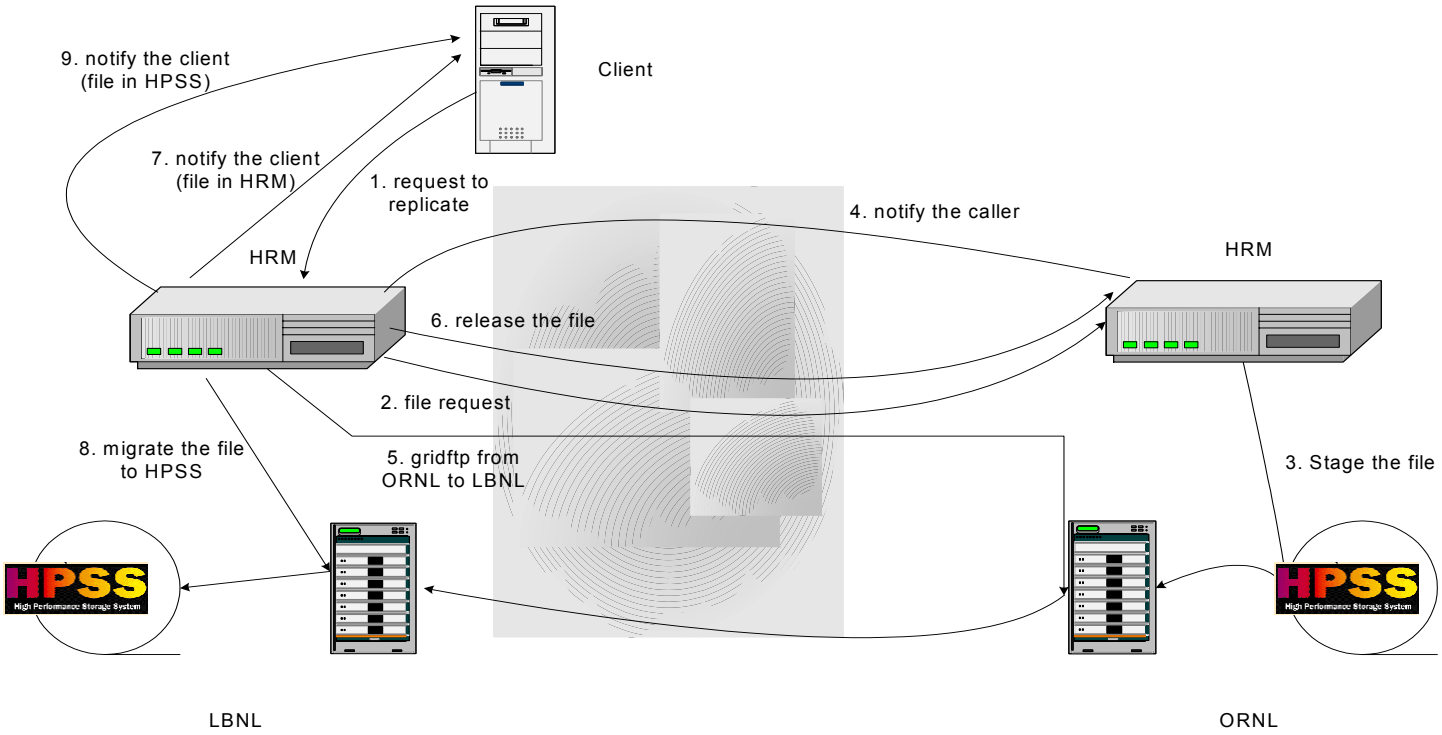
File replication using HRMs
from ORNL-HPSS to LBNL/NERSC-HPSS

Alex Sim, Arie Shoshani, Junmin Gu
Jan. 31, 2002

We have successfully replicated several files from ORNL HPSS to LBNL/NERSC HPSS using HRMs and HRM client commands, over Globus toool kit 2.0 beta. The HRM setup is shown in the following picture, and the steps are numbered. The client program used was new HRM command-line program (see specification at: http://sdm.lbl.gov/srm/documents/HRM.command.line.specification).



We ran total 9 runs, and only 1 run finished successfully (figure 1). We could not finish the other 8 runs because of a bug in the gridftp globus-gass-copy library (reference to the emails to gt2 mailing list). The Globus team is currently addressing this problem (Joseph Link from Globus team thinks that this is due to a race condition involving the completion of storing the data vs. receiving the confirmation from the ftp client library)

Next, we describe a 20-file replication test that completed successfully.

In Figure 1 below, constructed from logs the two HRMs generate, we show the progress of each file being replicated, from the time the request is made, through staging a file from HPSS at ORNL, transferring the file, and archiving the file at HPSS at NERSC. The X-axis represents time, and the Y-axis denotes the steps of the file replication process as shown in the following table:

1	File request is made by the client to LBNL-HRM and placed in the service queue
2	File is serviced by the LBNL-HRM for getting file
3	File request is accepted for service by the ORNL-HRM
4	File staging request is made by ORNL-HRM to ORNL-HPSS
5	File staging completed and cached into the ORNL-HRM managed disk
6	LBNL-HRM received callback from ORNL-HRM that the file is ready
7	LBNL-HRM initiates a Gridftp transfer from ORNL-HRM managed disk to LBNL-HRM managed disk
8	Gridftp completes
9	File release command sent by LBNL-HRM to ORNL-HRM
10	File archiving request to LBNL-HPSS is placed LBNL-HRM service queue
11	File is serviced by the LBNL-HRM for archiving
12	File archiving start from LBNL-HRM disk to LBNL-HPSS
13	File archiving to LBNL-HPSS completes, and the client is notified
14	File is released by the client

The X-axis denotes the time in the form of HHMMSS (e.g. 130500 = 1:05:00 pm)

All 20 files in figure 1 come from ORNL's production HPSS (they are ESG data files in Gary Strand's directory). The file sizes are around 350-390 MBs. All the files were transferred successfully and archive to LBNL's production HPSS.

We set the number of concurrent pftps to 5 for both staging from ORNL-HPSS and archiving into LBNL-HPSS. This limit was set in order to avoid flooding HPSS with pftp requests, and disrupt HPSS's normal operations.

In this run, we have used only 3 concurrent Gridtps (trying to avoid the gridftp bug mentioned above), and TCP/IP buffer size 1,000,000 and 2 parallel streams were used per gridftp.

From step 5 to step 6, there is long delay towards the latter part of the run. This shows that files staged at the ORNL end were waiting to be transferred. The reason for this delay is in part because of the network speed, and in part because we limited ourselves to 3 gridftps. While we observed in other tests a transfer rate of 11 MBytes/sec for a single file, the average transfer rate in this test was about 3.88 MB/sec (384 MB per file X 20 files, transferred in 33 minutes – see graph for begin and end of file transfer in steps 7-8).

Another interesting point to observe was the difference between staging speed at the ORNL-HPSS, and archiving speed at the LBNL-HPSS end. This is shown in steps 4-5 and steps 12-13, respectively. The ORNL end performed faster because the ORNL host

(sleepy.ccs.ornl.gov) has faster connection to the ORNL-HPSS than LBNL host (dm.lbl.gov) has to LBNL-HPSS.

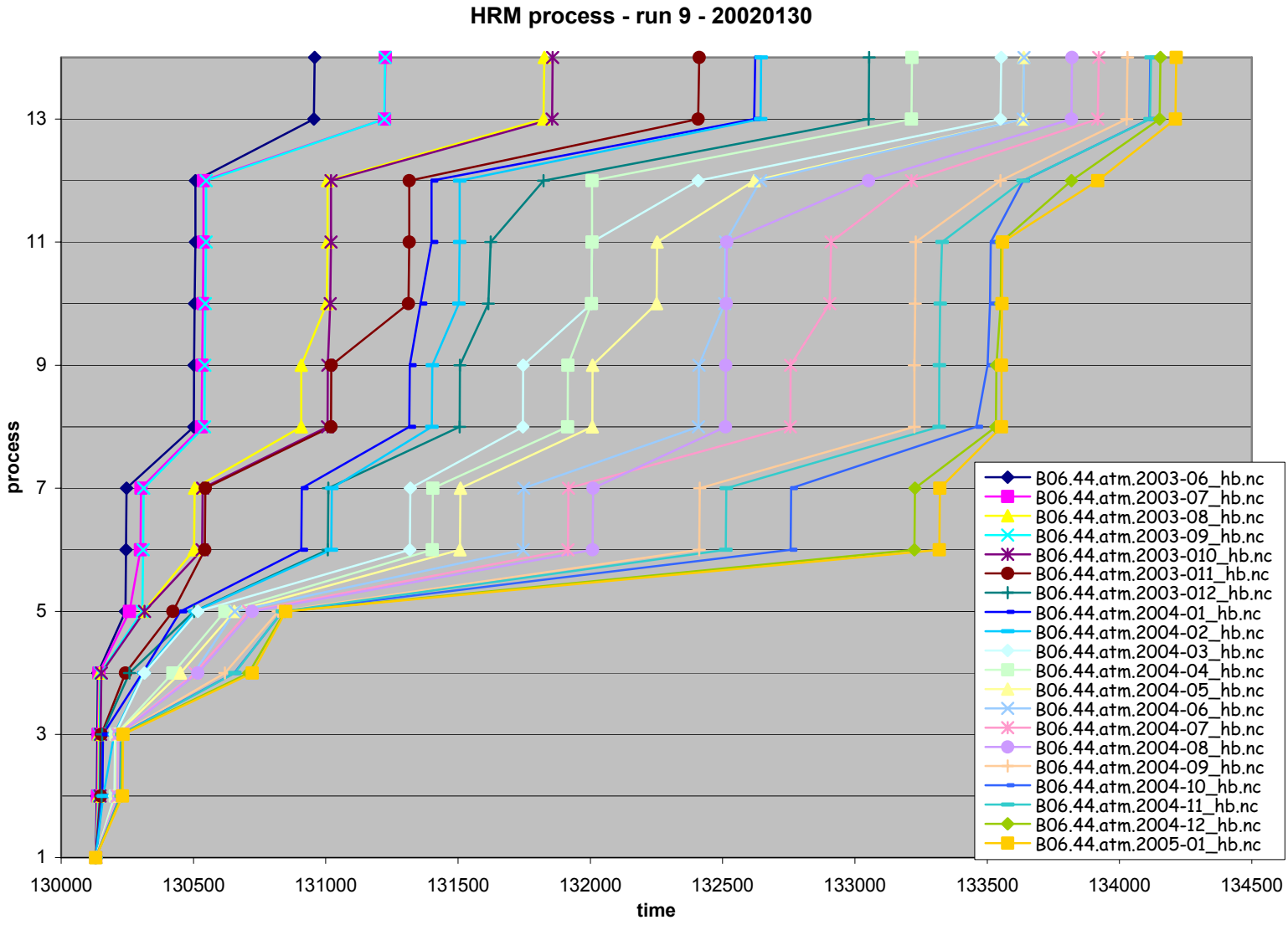


Figure 1: HRM process - RUN 9

Next, we show the graphs for one of the tests where one of the file transfers did not complete due to the bug in the globus-gass-copy library mentioned above. This test is also a 20-file replication test similar to the test shown in Figure 1. The difference is that in this run, we have used 20 different source files, 5 concurrent Gridftps, the TCP/IP buffer size was 500,000 and 2 parallel streams were used per Gridftp. The file transfer that did not complete is B06.44.atm.1996-08_hb.nc, shown in the graph in the red line with diamond connectors.

Interestingly, the average file transfer in this test was similar to the previous test (about 3.69 MBytes/sec from 342.9MB per file X 20 files, transferred in 31 minutes). It seems that increasing the number of concurrent Gridftps from 3 to 5, and using half buffer size did not affect average performance.

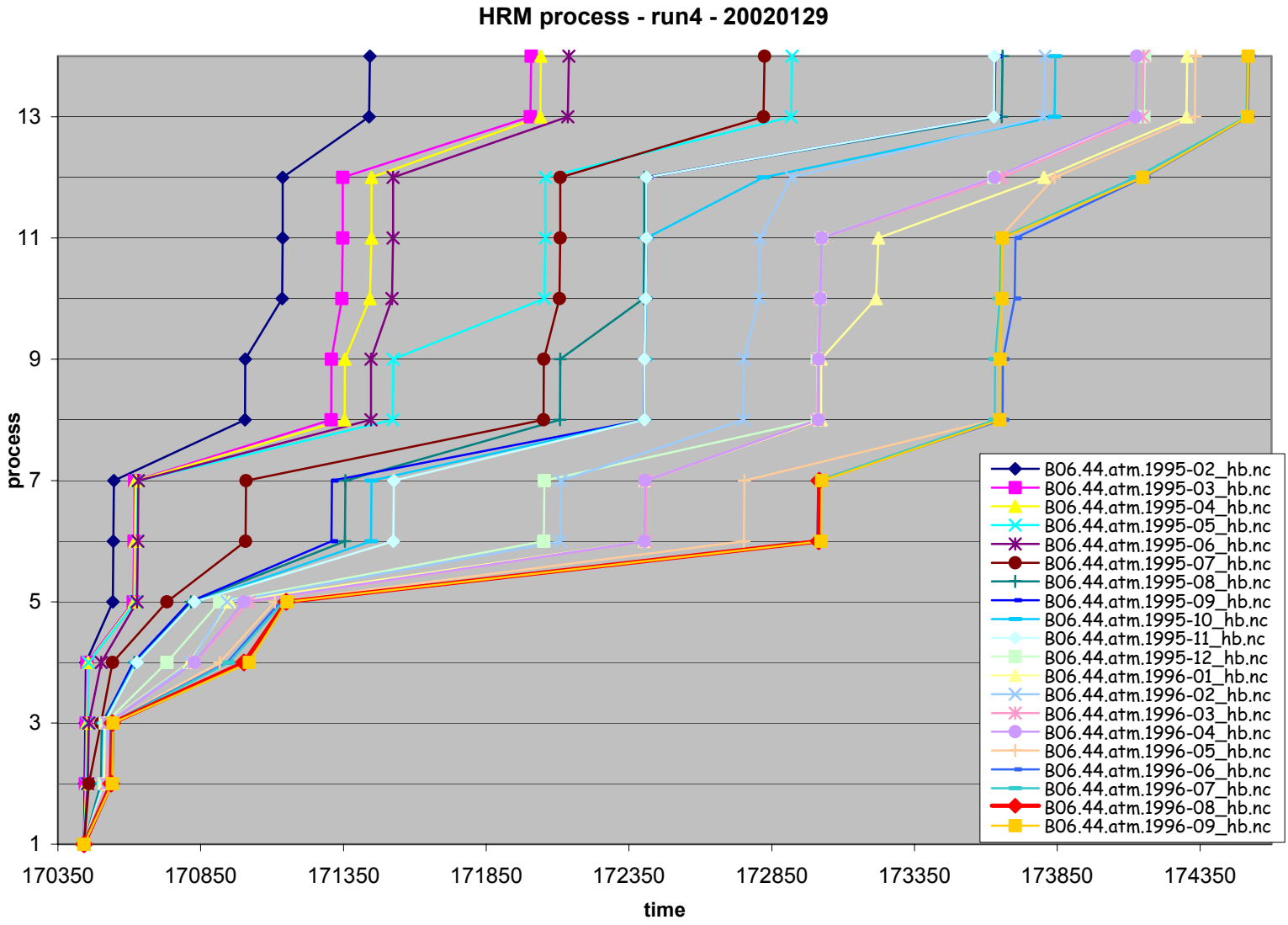


Figure 2: HRM process - RUN 4